



**RAYPACK®**

# Software Packaging

Getting Started with  
RayPack  
6.5

RayPack is part of RaySuite.



**Copyright © Raynet GmbH (Germany, Paderborn HRB 3524). All rights reserved.  
Complete or partial reproduction, adaptation, or translation without prior written permission is prohibited.**

## Getting Started with RayPack

Raynet and RayFlow are trademarks or registered trademarks of Raynet GmbH protected by patents in European Union, USA and Australia, other patents pending. Other company names and product names are trademarks of their respective owners and are used to their credit.

The content of this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Raynet GmbH. Raynet GmbH assumes no responsibility or liability for any errors or inaccuracies that may appear in this document. All names and data used in examples are fictitious unless otherwise noted.

Any type of software or data file can be packaged for software management using packaging tools from Raynet or those publicly purchasable in the market. The resulting package is referred to as a Raynet package. Copyright for any third party software and/or data described in a Raynet package remains the property of the relevant software vendor and/or developer. Raynet GmbH does not accept any liability arising from the distribution and/or use of third party software and/or data described in Raynet packages. Please refer to your Raynet license agreement for complete warranty and liability information.

Raynet GmbH Germany  
See our website for locations.

[www.raynet.de](http://www.raynet.de)

# Contents

<b>Get Started with RayPack .....</b>	<b>4</b>
How to set up your Packaging Environment .....	6
Architecture Recommendations .....	7
Prerequisite Hardware .....	9
Prerequisite Software .....	10
Product Activation .....	11
Packaging process recommendations .....	16
How to pick the right packaging method .....	18
Original software resources .....	19
Target Definition .....	19
Packaging Environment .....	21
How to Prepare Packages with RayPack .....	22
Definition of system settings .....	22
Creating a new MSI Package .....	24
Creating a new MST Transform .....	25
Creating an MSI Package From a legacy setup .....	26
Editing an existing MSI Package .....	27
Tailoring an MST for an MSI .....	29
PackRecorder Tutorial .....	31
PackTailor Tutorial .....	40
Need help? .....	44

# Get Started with RayPack

RayPack is a framework for the creation and management of software packages. It supports a broad variety of package formats, target operating systems, and deployment systems.

The RayPack components allow enterprises to implement well structured processes which control package creation, manipulation, import, export, validation, storage, and deployment.

In order to be able to manage these processes, RayPack utilizes integrated framework components along with external connectors, which enable interoperability with market leading workflow, software deployment, and data storage systems.

This Get Started Guide is designed to assist you during your first steps with RayPack.

## How to setup your Packaging Environment



[Architecture recommendations](#)



[Prerequisite hardware](#)



[Prerequisite software](#)



[Product activation](#)



[Packaging process recommendations](#)

## How to pick the right Packaging method



[Original software resources](#)



[Target package format](#)



[Packaging environment](#)

## How to prepare Packages with RayPack



[Definition of system settings](#)



[Creating a new MSI package](#)



[Creating an MSI package from a legacy setup](#)



[Editing an existing MSI package](#)



[Tailoring an MST for an MSI](#)

## Tutorials



[PackRecorder](#)



[PackTailor](#)

## Need Help?



[Request RayPack support](#)



[Join the RayPack community](#)



[Contact your Raynet sales representative](#)

## Additional Information

Visit [www.raypack.net](http://www.raypack.net) for further information regarding the product and current community incentives. It's also recommended taking a look at additional resources available at the Knowledge Base: <http://knowledgebase.raypack.net>

Raynet is looking forward to receiving your feedback from your RayPack experience. Please contact your Raynet service partner or use our [Support Panel](#) to add your ideas or requirements to the RayPack development roadmap!

# How to set up your Packaging Environment

Packaging at its best is based upon best practice recommendations. Read the following chapters to find out how a basic packaging environment should be set up:



[Architecture recommendations](#)



[Prerequisite hardware](#)



[Prerequisite software](#)



[Product activation](#)



[Packaging process recommendations](#)

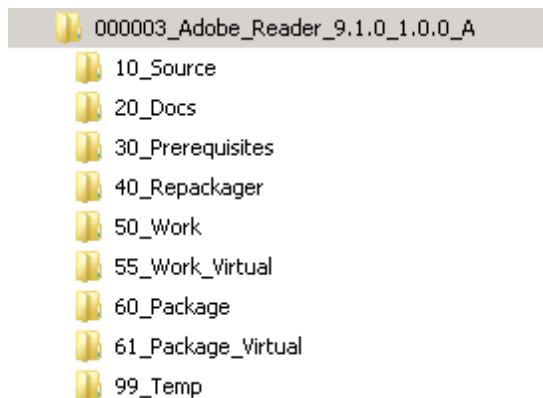
# Architecture Recommendations

Your packaging system needs to fulfill a certain set of requirements:

## Data Store

Each packaging job is a project, including basic (software) material, process steps, documentation, and a well designed target definition. The better a repository supports the organization of materials and information, the easier it is for the packaging team, which might very well solely consist of one user alone, to maintain efficient packaging procedures and high quality results.

Even though each packaging task has its own specifications, it is recommended to stick to a common handling as far as possible. The re-usage of templates and structures saves time and reduces error rates. Therefore, it is recommended to implement a standard repository structure, with separate folders for the different types of packaging media. The screenshot below shows one way of organizing media for a software packaging project within a data store:



The picture above shows an example of directories per packaging project for a file system based data store. The data store can be local, or established on any kind of network share. If it is intended to work on projects in a packaging team, a shared location all team members can access is most suitable. Depending on the requirements it might become handy to go a step ahead and manage the files within a database.

## Packaging Tools

Once the data structures are organized, it is time to think about the packaging process itself. Within reach of the data store, set up the packaging environment. This is the place to provide all the little (or not so little) helper tools required for successful packaging: A package editor, maybe a capture tool, some process organization support, quality assurance applications, and so on. These tools may be integrated into one handy framework, such as RayPack, or spread into several smaller "one trick" applications from one or more vendors. Check out the RaySuite for fitting applications, e. g. [RayEval](#) for package evaluation, or [RayQC](#) for quality assurance and testing support.

The exact set of packaging features required depends on the raw software material which is available at the start, the target format that is aspired, the size of the packaging team and the customers for who the work is done. Clarify the following questions:

- Is the basic software available in an MSI based format, or is it necessary to transform legacy setups in the first place? Transformation usually requires RayPack's PackRecorder.
- Which target operating systems need to be supported? RayPack includes the PackDesigner for Windows by default.
- Are MSI or MST format the target formats or is the objective a virtualized package? RayPack supports MSI and MST as target formats by default. Add the virtualization module to design App-V, ThinApp and other virtual package types.
- Is the work basically done by one individual, or does teamwork needs to be coordinated? Think about the benefits of implementing a central PackServer, with one central database for all packagers. Systems like RayFlow make it easier to coordinate work-flow steps from the initial customer package order until final delivery within spread team situations.
- How is the customer structure? Are packages prepared for one company, or are they delivered to several customers? Multi-client capability with separated storage organization will support convenient client management.

Please refer to a Raynet sales representative to get an overview of available product editions and additional modules to set up an individual packaging system. Whilst RayPack is a fully functional RayPack Studio, the proper edition selection tailors it right to individual requirements.

## Packaging Factory

Professional packaging depends on several dedicated machines (see [Prerequisite hardware](#)), which are configured for package manipulation, installation capturing and package testing. All those machines may be spread over several networks, be implemented on physical and virtual machines, and handled by a group of specialists. Prepare to adjust your individual packaging factory to match the requirements of your business and IT environment.

---

# Prerequisite Hardware

## Packaging Machine

The process steps of a common packaging project require at least a packaging machine running the application framework. Install the RayPack framework here. The data store for media management might also reside here. If the data store is placed on a shared network resource, make sure that it is accessible from the packaging machine.

Make sure your test and capture machines (see below) are as well within reach of the packaging machine. It's easier to transfer package material directly from one machine to another than to depend on crutches such as USB sticks or the like.

## Test Machines

A test machine is used to run installations and uninstallations during the packaging preparation and quality assurance. The test machine should be configured to be as similar as possible to the productive environment the packed software is designed for. If there are several target scenarios, it is recommended to test the package on all of them. Nowadays test machines are usually virtual machines, equipped with predefined sets of reusable images and status snapshots.

Testing includes installing the built package, checking for basic functionality, and also checking the system state after uninstallation for unwanted relicts. A well designed package does not leave any undesired footprints once its uninstalled.

## Capture Machine

If there are intentions to use capturing functionalities during the packaging process, another machine type is needed. This one should be a blank installation of the target operating system, extended with the required set of prerequisites (such as runtime environments) for the installation of the application that is to be packaged.

Based on system snapshots before and after installations, RayPack's PackRecorder calculates the delta between those two states. Therefore the differences have to be exact towards the recorded differences between the system states before and after the installation that is to be observed. If the capturing machine has other software installed, there might be issues and conflicts in resource usage, since every application installation brings along a varying set of resources that are integrated into the machine's system.

Imagine a `Sample.dll`, which exists in the initial snapshot taken from the system, because it was installed along with an application that is not part of the prerequisite set for the application that is about to be packaged. Now the application that is about to be captured is installed on the same system, requiring the `Sample.dll`. If it is already available, the installation will not change the system state in this point; the second snapshot taken after the installation will include the

.dll as well. Comparing the initial and post-installation snapshots will not reveal any change in the system state regarding this .dll file. Therefore, the delta file of differences will not include this specific .dll. If this missing resource is not detected, the following packaging process might very well lead to an erroneous result. If the .dll is part of a required prerequisite there is no major issue, since the package is intended to be placed into such a system.

Think of access attempts to the registry, to .dll's and directories as "background noise" caused by applications running on a machine. The capture machine should run with the little noise the possible, to make it easier to actually "listen" to the applications installation activities. Capturing results can be focused to specific system parts by exclusion lists, but the cleaner the capture machine is, the less need for manual correction and manipulation is required as capturing pre- and postwork.

By the way, using the term "machine" does not mean that there is a need for several physical machines for repackaging. Modern packaging factories are mostly virtualized, since it's way more handy to work with several virtual machines, instead of maintaining a broad variety of actual hardware machines.

## Prerequisite Software

### Operating Systems

Since your packages are designed to be installed on specific operating systems, the packaging results have to be tested on those operating systems. If capturing is utilized, best practice is to perform it on a blank target operating system. Therefore, maintaining licenses and ideally prepared (virtual) machines as well is a must have for high quality packaging results.

### Virtual Environments

As already mentioned, it is usual to virtualize testing and capturing machines. If the maintenance of a park of physical machines is not wanted, set up a virtual environment. Either just a bunch of loose images, or a more sophisticated solution, based on ESX, Hyper-V or any other common server technology. Preparing standard images for repeated usage does not only reduce your setup time per packaging job, but also leads to more reliable and comparable results. Reproducing errors during evaluation and quality control phases may take noticeable more time on even only slightly differing machines. Therefore it is recommended to utilize virtualization to make sure that identical sources are used where uniformity is required.

### Application Media

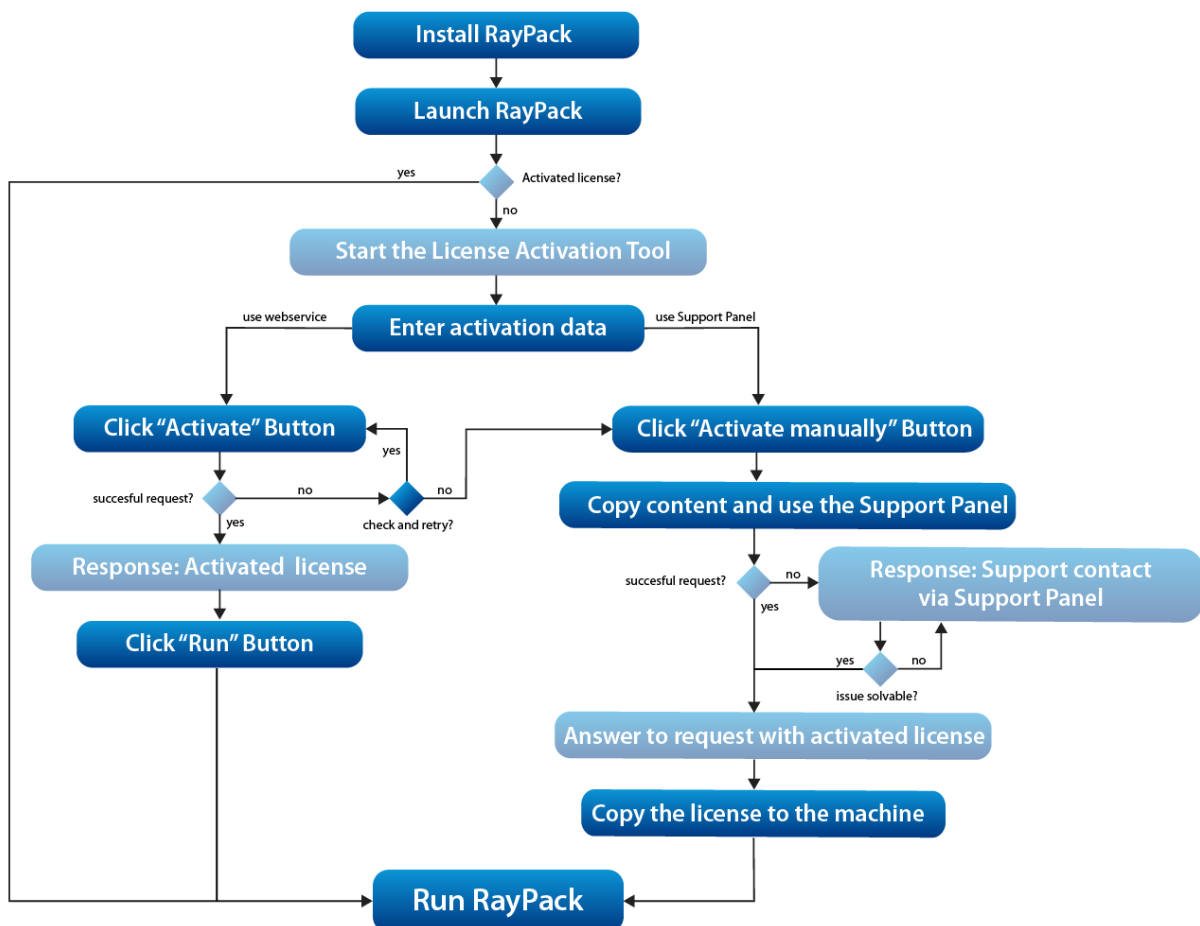
Packaging starts with the incoming software material. Nowadays world of software formats beholds a jungle of requirements. Not every vendor delivers standard compliant software, and for sure some packages that are delivered will be erroneous. Before starting to manipulate the original sources, it is of vital importance for a packaging job to check them for completeness regarding licenses, prerequisites, optional add-ons, and so on. Evaluate the raw material by installing and uninstalling it on a test machine. If a client does not deliver an installation

handbook or a detailed target package description, prepare it and go for client approval. Keep a copy of the original sources for later reference. Make sure that the desired packaging result can be achieved with the given resources. Please keep in mind that it is part of packaging Best Practice to avoid changes to vendor MSI's if not absolutely necessary. In most cases it is absolutely sufficient to add transforms to the original material, which allows to make sure vendor support can be contacted in case of issues with the original installer material.

## Product Activation

After RayPack has been installed on a machine, an activation process has to be executed as soon as the application is started for the first time.

There are two possible procedures for activation: [Direct online activation via web-service](#) or [support driven activation via the Raynet Support Panel](#). The second procedure is designed for environments without direct internet connectivity for the device.



Prepare the following information before the activation tool is invoked:

- **Order Number**

The order number is handed out with the installation resources or with the delivery note. It is

unique for every purchase of RayPack licenses.

- **Company name**

The name of the company that ordered the RayPack licenses.

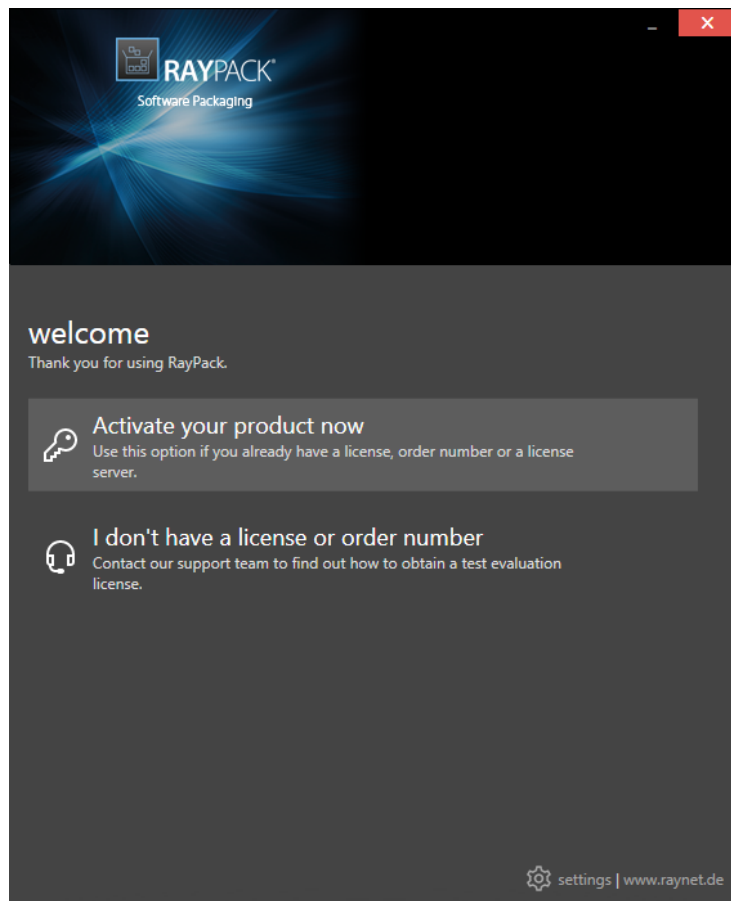
- **User name**

The actual name of the user who activates the product.

- **Email address**

The mail address of the user who activates the product (for consultation in case of support requests)

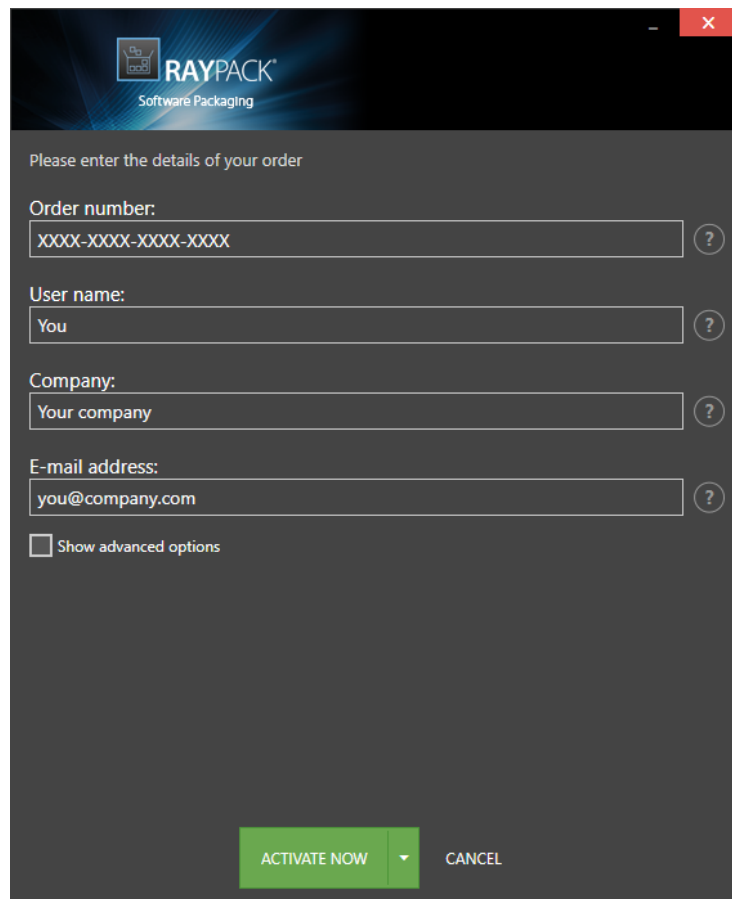
The **hardware Id** for the device is automatically generated by the License Activation Tool. It is required to establish the unique connection between the machine and the RayPack license. If the activation is done via email, the order number and hardware Id have to be sent to the Raynet support team in order to be able to process the product activation request.



Please choose the preferred activation procedure and follow the steps as described below:

## Online Activation via Web Service

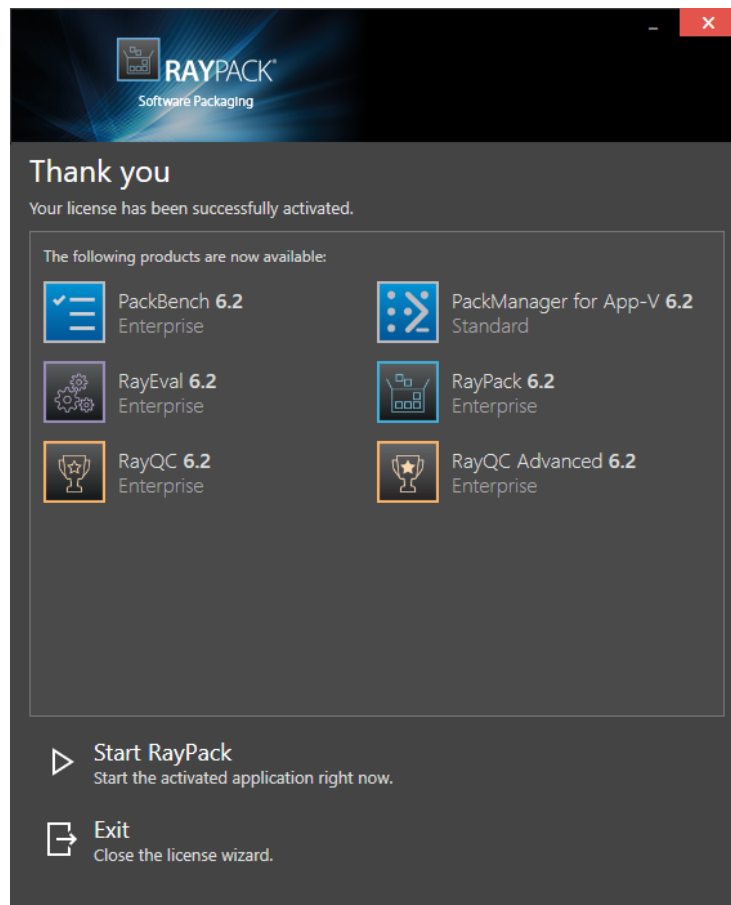
1. Enter the required information into the form fields of the License Activation Tool. The **Order Number** and **e-mail address** are required, and all other information is optional (but recommended).
2. Please remember to check the **"Transfer the license"** checkbox if you intend to replace an already existing license with a new activation. They can be found under advanced options checkbox.
3. Click the Activate button to **send the entered information** along with an automatically determined machine identification string (the hardware Id) to the activation web service.
4. If any field receives a red border around it indicates that its value is not accepted as a valid one. For example, the order number may not be empty, and the e-mail address must be in a valid format.
5. **Wait for the web service** to evaluate the activation request and send a response:



## The Activation Request is Valid

A license file is generated automatically and loaded into the RayPack program installation directory.

The Run button becomes available when the RayPack installation is successfully completed.



5. Click the **Run** button to launch RayPack, and automatically close the license activation tool.

## The Activation Request Could Not Be Evaluated

This happens whenever

- The order number is invalid  
Please verify if the entered order number is correct, and try again.
- All licenses purchased with the entered order number have already been used for activation  
Please contact your sales representative to order additional RayPack licenses. If it is needed to transfer a license from one machine to another, activate the "Transfer license" checkbox. By doing so, RayPack should no longer be used on the system that originally activated the license.



### Be aware:

Raynet intends to make product activation as convenient as possible. Nonetheless, no matter if activation is executed by web service or by support contact, Raynet saves the transferred information. If any kind of license usage is detected, that does not comply

---

with the End User License Agreement, Raynet reserves the right to take legal measures.

---

- A connection to the web service could not be established  
If there are temporary incidents that prevent a stable connection, try to reach a remote website or web service. If this fails as well, please check your internet connection in general.  
If the connection to the internet is stable, try activating again later, or use the support driven email activation procedure as described in the next section.

## Support Driven Activation Using the Raynet Support Panel

If the machine that hosts the RayPack installation which is about to be activated does not have a direct connection to the internet, the support driven activation process is required.

This procedure involves the Raynet support team as it initiates the generation of the license file by using your order information and the identification string of the device on which RayPack is installed.

1. Follow the same procedure as in the previous section, but choose the **ACTIVATE MANUALLY** button instead. Please note that all fields are required in order to perform an activation using our [Support Panel](#).
2. Use the **Print** key or the snipping tool to **copy the generated mail content** to a device with a mail client and internet connectivity.
3. Open a **License File Request** in our [Support Panel](#).

### The Activation Request is Valid

The Raynet support team generates an activated license and attaches it to the ticket response.

5. Save the **license file** to the **RayPack program installation directory**.  
(Typically something like `C:\Program Files (x86)\RayPack\`)

As soon as this activated license file is available, RayPack is fully activated and can be launched.

### The Activation Request Could Not Be Evaluated

The Raynet support team will contact you immediately to find the reason for the failed request. Please prepare your order information and hardware Id to compare the information that arrived at the help desk with your actual data.

---

**Be aware:**

Raynet intends to make product activation as convenient as possible. Nonetheless, no matter if activation is executed by web service or by support contact, Raynet saves the transferred information. If any kind of license usage is detected, that does not comply with the End User License Agreement, Raynet reserves the right to take legal measures.

## Packaging process recommendations

As already mentioned before, packaging is way more than simply manipulating some package properties. Before starting to go for actual package designing, make sure that the target installation scenario is well defined, the basic material is flawless and complete, and there exists a clear conception of quality criteria the package has to fulfill.

Wrapped in the standardized Enterprise Application Lifecycle Management process, the steps for the realization of a package request might very well come close to a workflow similar to this one:

1. **RayFlow:** Incoming package order from the customer
2. **RayEval:** Evaluation of packaging requirements, creation of installation description
3. **RayPack:** Packaging procedures (tailoring a transform, repackaging legacy formats, designing complex packaging solutions, package validation)
4. **RayQC:** Quality control tasks regarding the delivered target package
5. **RayFlow:** Package acceptance by the customer
6. **RayManageSoft:** Deploying the package according to customer standards

Since packaging is only one part of the EAL process (=Enterprise Application Lifecycle), the actual full cycle requires prior steps, such as recording inventory to be able to gather target machine requirements, licensing status, and the like.



Depending on your individual packaging environment, the tools you utilize to master process steps may vary. The illustration above shows a typical setup for frameworks which operate on Raynet standard applications and services, such as [RayFlow](#), [RayEval](#), [RayQC](#), or [RayManageSoft](#). However, RayPack is a framework you can implement in combination with any kind of third-party tool.

Experienced packagers base their work on a core set of standard procedures, and adjust them towards the specific requirements of each individual packaging job and project environment: No matter which format or operating system is the target - no package should go out to the productive environment without decent quality checks. However, which tests have to be applied may vary. The same is to be said about package deployment. It is a usual enterprise requirement to create a package that is installed silently, meaning without the need for user interaction during installation. Beyond that common ground, the market leading deployment systems do not share a common standard for deployment management. Depending on the operative deployment method, each custom software package has to be adjusted to be able to flow smoothly through the individual deployment process.

## Naming Conventions

In order to organize package resources, it is highly recommended to apply a naming convention for projects and packages. There are basic properties which are available for each package project. Using them to organize the storage hierarchy within the data store provides quick orientation and package identification at a glance. Sticking to a structured convention also enables automated task execution and rule based data extraction.

Typical elements of a package naming conventions are:

- Local package id

If using workflow management tools, such as RayFlow, it is common to define a project specific, locally unique job id. In order to quickly identify the package that belongs to a certain RayFlow job, it is convenient to use the id within the package project name.

- Vendor name, software name, software version  
The combination of these three properties definitely identifies the basic software. Make sure to use a standard version convention to reflect major, minor and patched versions.
- Package type  
The target package type is decisive for many operational steps, therefore a type identifier within the project title provides essential information about the project without the need to open it's documentation or project file. Make sure to define a standard identifier system to use throughout all packaging projects.
- Source type  
Add an identifier reflecting the source media type. Is it a native MSI file, as provided by the original vendor, an original legacy installation, an already wrapped or repackaged MSI.

Again, it is not necessary to apply a naming convention, but defining one will surely lead to benefits regarding the organization level of the packages and the level of automation that can be applied on the stored package media.

## How to pick the right packaging method

Packaging is not a straight forward, uniform process. Picking the right packaging method may not only save a serious amount of working time, but also enable the ability to reach the desired goal in the first place. One has to adjust the applied methods regarding job specific target formats, client operating systems, and available software sources.

Make sure to gather decisive information before starting to actually work on any package. Read on to get some hints on how to decide for the packaging method that fits specific job situations:



[Original software resources](#)



[Target definition](#)



[Packaging environment](#)

## Original software resources

The software resources may be delivered in uncountable variety of formats. It might even come down to just a bunch of files and a list of destinations where to put them, and how to configure the environment. Wouldn't it be nice to be able to have one single packaging method that covers them all? One ring to rule them all, one ring to find them, one ring to bring them all and, well, build them?

Here are good news: Within RayPack there is such a tool - the PackRecorder. Its technology of taking two snapshots, one before and another one after installation, allows to manipulate installation and configuration steps in any desired way. Therefore, the PackRecorder can be used on any challenging software material, and build standardized packages from it.

But, and of course there is a but, otherwise it would not be necessary to read this, capturing is a process with high requirements regarding preparation and post-processing. To make sure that the delta file contains exclusively those changes that are of relevance for the final package, there is a serious amount of care that has to be applied to the preparation of the snapshot machine, on exclusion lists, and possible collision conflicts on the target system.

Therefore it is not recommended to rely on the PackRecorder in every case, even though it would work. Think of the PackRecorder as the final arrow within the quiver, and check other possible, specialized solutions first before pulling it.

If equipped with an MSI based file, don't put all the extra fine-tuning work load on one packagers shoulders, but import the file into the PackDesigner. PackDesigner offers direct access to freely manipulate the tabular package foundation, as well as a sophisticated user interface to support the user with guided procedures and assisting control structures. Especially those vendor MSI resources, which are assigned for MST extension, are usually no candidates for repackaging, but should be handled with either PackTailor or PackDesigner and their transform management features.

### To sum up:

1. If an MSI is available, and the UI sequence contains switches for all options required to manipulate, take a tour through the PackTailor to get the job done with a quick MST.
2. Check the original resources for importability into PackDesigner. If it can be handled directly, and there are no additional needs that cannot be met otherwise, use PackDesigner and enjoy the feast of handy functionality it offers.
3. Only go for the capturing detour if the options named above cannot be applied on the available resource material.

## Target Definition

### Target Format

---

What kind of task does the target package have to achieve? Does the package need to be installed as a real application on a physical machine, will it exist as a virtualized application, or run in a virtual environment? Is it necessary to update or patch an existing application?

RayPack supports the generation of virtualized packages, which only pretend to be a physically installed application, but actually include instructions on how to emulate installation aspects such as environment variables, registry keys, file system resources, and the like.

The appropriate type of application virtualization format depends on the customers infrastructure. If an Application Virtualization Management Server is implemented, the customer might very well request an App-V package, which contains all application streaming instructions and data required to stream the included software via such a management server.

Other scenarios require software to run from an USB stick, or any other portable storage media, without the ability to extract files or settings to the target system. These cases require package formats such as ThinApp, which encapsulate application packages for totally portable usage.

If the basic material has the same format as the target package, it is usually the easiest way to import the package into the PackDesigner, execute the required changes, and export the resulting set of operational instructions. Add MST files to MSI resources, or transfer from one file type to another, for example by importing an MSI file, and exporting it into the App-V format.

However, in some cases it is necessary to leave the original material as is, and just extend it with a transform, for example if an application is to be patched. For MSI files this is the recommended method, especially if the original application is already installed on the target system. It does not make sense to reinstall the whole application, putting existing user data and settings at risk, but fix a certain issue using the repair functionality of MSI based applications. In those scenarios it is the safest way to create an MST file, which contains instructions on how to manipulate the installed application data safely.

Another scenario for MST creation would be the fact, that an MSI file with original resources is available, providing all required options for manipulation directly editable for users during the installation routine. Whatever a user may decide during a manual setup, may be bundled into an MST file by PackTailor. Simply create a new PackTailor project, run through the simulation of the UI sequence, and enjoy the fact that with a decent base MSI, your target MST is potentially only a few clicks away.

## Target Environment

Software packages are usually dedicated to one or more specific target environments. The environment definition includes aspects such as operating system, corporate network and security factors, co-existing applications, update schedules, and language settings.

Creating packages with software that represents the client within a client-server application architecture, the required location and authorization data has to be provided to make sure that the installed package can be used straight away without further configuration by the end-user.

Make sure to agree on an exhaustive operational environment definition with the customer in

the first place and evaluate the requirements regarding his feasibility before any packaging activity starts. RayPack will offer support with package validation on several levels, including Logo checks for Windows operating systems, ICE validation, and Package Signing. Depending on the required OS coverage, ensure to create a package that passes these validation steps in the context of the targeted systems.

Once knowing what environment to aim for, some target formats, and therefore packaging methods, might turn out to be inapplicable. It does not help at all to prepare an MSI package if the customer runs his IT landscape based on virtualized desktops. On the other hand the availability of virtualization options depends on application requirements, for example regarding prerequisite drivers, which might very well make it impossible to virtualize an application.

## Target Installation Type

Another aspect of deciding for the right packaging method is the desired installation type. If a silent installation is required, but the original software material inevitably requires user input during the installation, PackRecorder has to be used first in order to capture those inputs, and wrap them into a silent package installation procedure. PackTailor might also do the trick, but whenever complex installation routines work with picky launch conditions, system state analysis, custom action based dialog management, and the like, simply tailoring the UI sequence usually does not work out sufficiently well.

Precapturing may also be required if the target installation demands advertised feature installation, and the raw material does not support it. It is possible to extend an MSI file with a transform to achieve advertisement, but in order to integrate it into a legacy setup or a virtualized package requires additional preparations.

## Packaging Environment

Packaging can be done in several environment structures, which differ not only regarding their localization, but also regarding the requirements they have to meet.

Packaging factories can be located **on-site**, with packaging staff working directly from inside the customers IT landscape. They can also be located **off-site**, communicating via tools such as remote connections, and web based workflow management and communication tools. Raynet for example usually applies **virtual** packaging factories, with each functional team segment, namely customers, evaluators, packagers, quality testers, and the like, located wherever their business requires. The benefits of cost reduction for establishing physical on-site packaging factories come at a price: a higher need for teamwork coordination. Therefore it is inevitable to organize team members and their tasks utilizing management tools, such as RayFlow.

Once having analyzed the customers IT landscape and the locations of the packaging factory team members, issues regarding security and network access availability might be encountered. Whilst most packaging tasks can be handled off-site, or with remote access to on-site packaging machines, some tasks require physical presence, for example whenever a highly specialized target environment cannot be emulated. Think of maximum security areas without network access, think of customized manufacturing lines, and so on. In these cases it might very well be necessary to install the packaging framework on a powerful notebook and travel on-site.

Especially the analysis of possible application conflicts and the preparation of virtual capturing machines has to be managed with the highest precision directly at the point of action.

Therefore it is recommended to analyze the individual job requirements first, and equip the packaging factory team with the required packaging tools. Prepare this thoroughly, no matter if working alone or in a voluminous team of specialists, spread all over the continents.

## How to Prepare Packages with RayPack

Once it has been decided what to do, it is needed to dive into the RayPack framework architecture, identify the correct process templates and use the matching packaging editors. The following chapters provide an overview of how to execute basic packaging tasks within PackDesigner and PackRecorder:



[Definition of system settings](#)



[Creating a new MSI package](#)



[Creating an MSI package from a legacy setup](#)



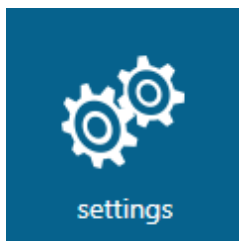
[Editing an existing MSI package](#)



[Tailoring an MST for an MSI](#)

## Definition of system settings

Before starting with the packaging, take the time to define essential system settings.



To do so, call the **settings** tile from the dashboard, or use the **settings** icon, which is always available from the menu bar at the upper right corner of the RayPack application window.

Open the packaging settings by clicking on the **profiles and settings** tab of the **settings** area.

The general settings view demands the definition of a standard folder for

packaging profiles. Profiles are permanently saved setting definitions can be prepared and activated on demand. It is recommended to keep them all within one central profiles folder. The currently active profile is preselected and displayed in a blue tile color in the **Profiles** tab. Whenever a new profile is created (by using the **Select or create another profile...** button right under the tile), it is automatically saved into the standard profiles folder. Select a shared network location to be able to apply RayPack settings profiles to several packaging systems at a time.

RayPack comes along with a predefined **Default** template, containing general settings that should work for most of scenarios and packaging projects. If creating a new settings profile, make sure to apply a speaking name for it, since as packaging projects go by it might be that a handful of specialized settings profiles will be developed and therefore later the right one will have to be picked intuitively at best.

Once a profile has been selected from the tile overview, its settings can be manipulated by using the five tabs:

- **Projects**, where basic templates and default saving location can be defined,
- **Repackaging**, where PackRecorder settings can be edited,
- **Designing**, where PackDesigner settings can be edited
- **Signing + tagging**, where package signing and ISO tagging can be defined,
- **Virtualization**, containing ThinApp and App-V settings.
- **RayFlow**, containing RayFlow URL settings

For details on the tabbed dialogue contents, open the specific tab and press **F1** to get a glimpse on the application help files for that area.

RayPack comes along with a feature called PackPoint. It is available for professional and enterprise edition users, allowing them to reuse a central repository of common resources (templates, merge modules, exclusions, predefined properties, etc.). Once established, PackPoint is an essential companion for the maintenance of high level teamwork and standardization measures in packaging factories.

In order to use the PackPoint organization structure, the storage location for the PackPoint has to be defined during the product instance setup. If a PackPoint is defined, the initial population with resources is executed when a RayPack application instance is launched for the first time. All core resources, which are appropriate for permanent maintenance beyond product upgrades and local instance endurance, are automatically put to the PackPoint location. The material preserved within the PackPoint location is the base for any synchronization with the local resources on the packaging machine.

Please refer to the *User Guide* for further details on how to work with the PackPoint, and make sure to follow the instructions provided during the product setup to use the right PackPoint option for the packaging machine.

# Creating a new MSI Package



## Be aware:

Remember to define [system settings](#) before creating or manipulating package contents with RayPack!

This chapter describes how to create a new MSI package from scratch. Instructions on editing MSI packages are available [here](#).

1. To create a new MSI package project, launch **RayPack**.
2. From the **dashboard**, call the **create a new project** tile.  
It is also possible to use the **FILE** button from the main toolbar, select **New**, and pick **empty project** from the provided activity options.
3. A selection screen is shown, allowing to specify the target path, language, schema and platform. Once the **Create new project** button is clicked, a project in RPP or MSI format will be created.
4. The **PackDesigner** is displayed, presenting the **YOUR PROJECT** overview.
5. Start to add contents to the packaging project now.

To do so, either click on one of the tiles within the **YOUR PROJECT** overview, or select the property to work on from the tree menu on the left side of the application window.

6. To temporarily save the project, click the **Save** or **Save as...** button, available in the **File** menu.
7. Once all required package contents are created, open the **Advanced View** by clicking its icon in the activity bar. The **TABLES** view presents a **VALIDATE** button, which allows the selection of several rule sets to apply.

This level of validation is quite superficial, since it does not give any feedback about potential application compatibility conflicts. Validation within the PackDesigner is a prequality assurance task, designed to reduce the reject rate between packagers and quality testers. The basic package validation is no replacement for decent quality checks later on!



## Note:

The compatibility testing is available in RayPack after the installation of RayQC Advanced module. Please refer to the *User Guide* for further details on how to work with the *Qualitycenter*.

8. If the validation revealed serious errors, handle them by extracting a fitting solution from the error description text. Revalidate until all critical errors are solved.

If the **validation** is successful, build the MSI file from the package project.

9. To build an MSI file, use the **Build...** button (available in the **File** menu) and select MSI as target format.

10. Wait for the save process to **finish**.

The MSI package is created, ready for further testing, and - finally - deployment.

## Creating a new MST Transform



### Be aware:

Remember to define [system settings](#) before creating or manipulating package contents with RayPack!

This chapter describes how to create a new MST transform to adjust a vendor MSI package. Instructions on editing MSI packages are available [here](#).

1. To create a new MST package project, first open the base vendor MSI.
  - From the **dashboard**, call the **open** tile.  
Or use the **FILE** button from the main toolbar, select **Open**, and pick **Windows Installer project** from the provided activity options.
2. The **PackDesigner** is displayed, presenting the **YOUR PROJECT** overview.
3. Start to add contents to the packaging project now.

To do so, either click on one of the tiles within the **YOUR PROJECT** overview, or select the property to work on from the tree menu on the left side of the application window.

It is also possible to apply a transform template now. To do so, click the **Transform** button available in the File menu, and pick **Apply changes from transform template...** functionality. The template file can be selected.

4. To save the changes as a Windows Installer transform file (`.mst`), click the **Save as...** button, available in the File menu. Pick **Windows Installer transform** from the provided activity option and select the target location of the new transform.
5. Continue with adjusting the package. Click the **Save** button, available in the File menu, to update the MST file (Note: The underlying MSI file is unaffected).



### Note:

To create a response transform, use the [PackTailor](#) wizard.

# Creating an MSI Package From a legacy setup

**Be aware:**

Remember to define [system settings](#) before creating or manipulating package contents with RayPack!

**Note:**

Some wizard steps are exclusively available for users that have activated the **Experts mode** for the Capture Wizard. To enable the Experts mode:

- Open the **SETTINGS** section, call the **Repackaging** tab. Then select the **WIZARD** tab and click on the Experts mode tile on the upper left hand side. The setting is permanent and will last until it is changed manually.
- Activate the Use experts mode checkbox on the first wizard screen. The setting is temporary and will not be considered for any later capture procedure. However, for the current session it overrides any definition from the overall settings profiles.

This chapter describes how to create a new MSI package from a legacy setup. Instructions on editing MSI packages are available [here](#).

1. To create a new MSI package project from a legacy setup, launch **RayPack**.
2. From the **dashboard**, call the **create a new project** tile or use the **FILE** button from the main toolbar, select **New**, and pick **repackaging** from the provided activity options.
3. Select **Capture a setup** from the new project dialogue.
4. The **PackRecorder Capture Wizard** is displayed.
5. Fill in the required information, and proceed from one step to the next by using the **NEXT** button, available in the lower right corner of the wizard dialogue window. Use the **BACK** button to return to an earlier step.
6. Within the **Options** step, assure fitting settings for file and property exclusions, for scanned system parts and the like. This step is not present during the Basic mode.
7. If the experts mode is active, it is possible to **select an already existing snapshot** instead of creating a new one. This is quite handy if the same clean machine is used over and over again for repackaging. Simply make one initial snapshot, and save the required time for fresh capturing each time the process of capturing with the PackRecorder wizard is executed.
8. The following step generates the **initial system snapshot**. Make sure that the system is as clean as possible at this point. Taking the snapshot might take a few minutes. The step is skipped if an already saved snapshot has been selected previously.
9. **Install your application** now. Apply command line options as required. If the experts mode

---

is active, it is possible to save the snapshot for later re-use.

10. It is also possible to make additional, manual **configurations to the system**, but keep in mind that all changes that will be executed now will be reflected within the final packaging project! This step is not visible in the basic wizard mode.
11. If a system reboot is required, initiate it now. The Capture wizard will automatically be opened after the reboot, ready to proceed.
12. Once the installation and configuration measures are complete, the second snapshot is due. If the experts mode is active, a dialog is displayed that allows to select an already existing snapshot instead of creating a new one. This is handy if the current wizard run is executed in the scope of analysis and uninstallation quality assurance, and not repackaging.
13. Proceed to the **second snapshot**. It is generated by applying the same options settings as were active for the first snapshot. Taking the snapshot might take a few minutes. The step is skipped if an already saved snapshot has been selected previously.
14. After the second snapshot is taken, RayPack automatically calculates the snapshot **delta**, and saves it as a `.rcp` file. On the final wizard screen, there is an option to save the second snapshot for later reuse if the experts mode is active.
15. Once the process is finished, decide whether to return to RayPack's home screen, or if the newly created capture project should directly be opened within the Capture Editor for fine-tuning.
16. Open the `.rcp` file from the **dashboard** (or by activating the checkbox for immediate edition on the last page of the capture wizard), using the **open capture project** tile.
17. Within the **PackRecorder editor**, manipulate the required package properties.
18. To build an MSI package from the capture project, click on the **FILE** item from the main toolbar. Select the **Build** item from the menu column on the left and activate the desired target format, which is Windows Installer for the generation of an MSI file.
19. Wait for the build process to **finish**.

The MSI package is created, available at the predefined outgoing package repository (controlled by the active settings profile), ready for further testing and - finally - deployment.

## Editing an existing MSI Package



### Be aware:

Remember to define [system settings](#) before creating or manipulating package contents with RayPack!

---

This chapter describes how to edit an existing MSI package. Instructions on creating new MSI packages from scratch are available [here](#).

1. To edit an MSI package, launch **RayPack**.
2. From the **dashboard**, call the **open project** tile, browse to the storage location of your MSI file, and click **open**.

Alternatively, select the MSI file from the list of recently opened items on the right side of the dashboard if it is available there.

Or use the **FILE** button from the main toolbar, select Open, and browse for the required package resources.

3. The MSI file is opened within the **PackDesigner**.
4. Start to manipulating the contents of the packaging project now.

To do so, select the property to work on from within the tree menu on the left side of the application window.

5. To temporarily save the project, click the **Save** or **Save as...** button, available from the File menu.
6. Once all required package contents are created, open the **Advanced View** by clicking its icon in the activity bar. The **TABLES** view presents a **VALIDATE** button, which allows the selection of several rule sets to apply.

This level of validation is quite superficial, since it does not give any feedback about potential application compatibility conflicts. Validation within the PackDesigner is a pre-quality assurance task, designed to reduce the reject rate between packagers and quality testers. The basic package validation is no replacement for decent quality checks later on!



**Note:**

The compatibility testing is available in RayPack after the installation of RayQC Advanced module. Please refer to the *User Guide* for further details on how to work with the *Qualitycenter*.

7. If the validation is successful, build the MSI file from the package project.

If the validation revealed serious errors, handle them by extracting a fitting solution from the error description text. Revalidate until all critical errors are solved.

8. To build an MSI file, use the **Build** button (available in the FILE menu) and select MSI as target format.
9. Wait for the save process to **finish**.

The MSI package is created, ready for further testing, and - finally - deployment.

# Tailoring an MST for an MSI

**Be aware:**

Remember to define [system settings](#) before creating or manipulating package contents with RayPack!

**Note:**

Make sure to use PackTailor with an MSI for an application that is not already installed on the packaging machine to achieve clean tailoring results.

This chapter describes how to create an MST for an MSI package by using PackTailor. Please refer to the MSI related sections to get information on how to [create](#) and [edit](#) MSI packages themselves.

1. To tailor a new MST for a specific MSI package, launch **RayPack**.
2. From the **dashboard**, call the **create a new project** tile.  
Or use the **FILE** button from the main toolbar, select **New**, and pick **response transform** from the provided activity options.
3. The predefined system settings are applied on the new project.
4. The **PackTailor** is displayed, presenting the **Target MSI** step of the Tailoring wizard.
5. **BROWSE** for the MSI that is the tailoring base.  
Activate the command line check-box if additional execution options have to be applied in order to achieve a UI sequence that is correct towards the target definition.
6. Click **NEXT** to proceed.
7. The **Additional transforms** step is displayed.  
If there are one or more transforms that needs to be added to the base MSI, use the ADD TRANSFORM button as often as required to add them.
8. Click **NEXT** to proceed.
9. The **Capturing user input** step is displayed.  
If advanced settings, such as the option to ignore the current system state or existing launch conditions for the installation, have to be configured, expand the advanced section, and activate the given check-boxes.
10. Click **START TAILORING** to initiate the UI sequence of the MSI. The installation runs as a simulation, the system state of the packaging machine used will not be changed during the tailoring process.  
Run through the UI sequence step by step and make adjustments to the available installation options according to the presented steps.

11. As soon as the UI sequence is finished, the **Results** step is displayed.

The tabbed view allows to browse the changes recorded regarding Features, Properties and Folders.

If any of the recorded changes should not be included into the tailored MST, deactivate the checkbox in the outer-left column of objects row within the changes table.



**Be aware:**

It is not recommended to manually adjust the changes if possible dependencies are unknown. Changes may depend on each other, and broken dependencies may lead to an invalid MST result

12. Click **NEXT** to proceed.

13. **BROWSE** to modify the name and location of the MST that will soon be generated.

If any transforms had been added to the tailoring process before, decide whether the resulting MST should include all transform instructions should be bundled into the one tailored MST, or if the original transform instructions should not be included in the resulting MST.

Additionally, decide whether the combination of original MSI and now tailored MST should be opened for review and fine-tuning in PackDesigner as soon as the MST creation is complete.

14. Click **FINISH** to start the MST creation.

15. The PackTailor wizard is closed automatically and a progress indicator dialog is displayed. Wait for the process to **finish**.

The MST is created, ready for further fine-tuning, testing, and finally deployment.

# PackRecorder Tutorial

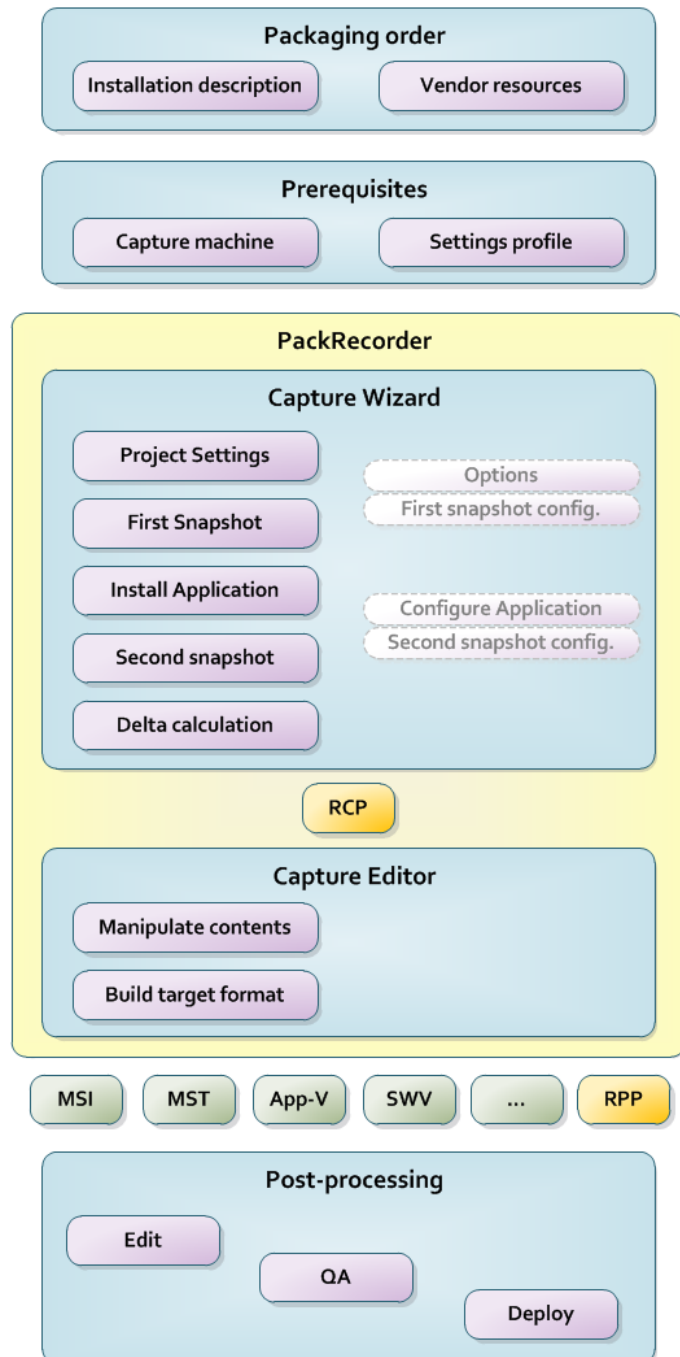
## What is the PackRecorder?

The PackRecorder is RayPack's recapturing tool that transfers non-MSI setups into a standard format which will serve as a base for further packaging tasks. PackRecorder uses snapshot technology to catch the system activity during installation and configuration processes.

Before the installation process begins, an initial snapshot is taken from a clean repackaging machine. At that point, the machine should reflect the target installation OS and should be prepared with the required prerequisites for the specific software (installation of Java Runtime Environment or .NET framework).

After the initial snapshot, the application that is to be repackaged must be installed. Additional configuration steps that must also be recorded, like defining specific user settings within the installed application, can be applied immediately after the installation is completed.

Once the installation and configuration steps are completed, the second snapshot is taken. Both snapshots are then compared to assess the disparities the installation caused on the system, which are then stored in a so called Delta file. This raw material is used to create the RayPack capture project file (.rcp), which can be opened and manipulated with the Capture Editor of the PackRecorder.



## What are the prerequisites for capturing snapshots?

Before any actual packaging activity, no matter if it is package design or repackaging, it is highly recommended taking a look at the settings section first. For details regarding settings, please refer to the **Settings Tutorial** within the Knowledge Base, or dive into RayPack's **Application Help** contents.

As mentioned above, capturing with RayPack is a snapshot based process. Therefore, the usual approach is to execute capturing on a machine that is as clean as possible with regards to previously installed software.

In order to achieve the desired repackaging results, there should be a detailed installation routine description prepared or assigned by the customer before the packaging project enters the active package design phase. These instructions also contain information about necessary configuration steps that follow the actual installation.

## Create a capture project file

To start a new capture project, use the create a new project tile from the RayPack home screen, or click on the FILE button, available from the main toolbar at the top of the RayPack application window, and select New from the options menu on the left.

The **new project** type is displayed in the dialogue window. Choose “Capture a setup” to open the Capture Wizard and use as a guide through the capturing process.



### Note:

The steps of the Capture Wizard depend on the Capture Mode defined within RayPack's Settings section:

- If the mode is set to *Basic*, the wizard will contain Project Settings, First Snapshot, Install Application, Second Snapshot, Finished.
- If the mode is set to *Expert*, the wizard will contain Project Settings, *Options*, *First Snapshot Configuration*, First Snapshot, Install Application, *Configure Application*, *Second Snapshot Configuration*, Second Snapshot, Finished.

The additional steps in the Expert Mode enable the manipulation of advanced options for this specific capture process. The standard settings for capturing remain unchanged by these adjustments.

The *Snapshot configuration* views are designed to decide whether new recordings or existing snapshot files should be used as base for the delta file calculation process. Whenever a fresh snapshot has been taken, the Experts Mode PackRecorder offers a button to save the recorded information as `.rcs` file (RayPack Capture Snapshot) for later re-using.

**Be aware:**

It is highly recommended closing all non-required applications and discontinuing dispensable processes and services running in the background before the Capture Wizard is initiated.

## Project settings

Define the **Application name** as project title. It is recommended to include the vendor name, as well as the application name and version into this value. If there is a dedicated project naming practice that is used, apply it at this time.

The default **Project directory** value is taken from the current active settings profile. Changes made within the wizard only affect this single repackaging procedure, so the value permanently stored within the settings profile remains unchanged. The Project directory automatically stores project files, and thus may require a significant amount of free disk space, depending on the size of the package and the number of target package formats intended to be generated. In order to make the project files available to a group or team of employees rather than an individual, it is recommended to set the project directory to a shared location within the network.

Browse the system to define **The main installer** that needs to be captured during the snapshot procedure. The user can change this value, and also add command line control options before the actual installation is executed. By this time, there should already be sufficient required sources on hand.

Activate the Use experts mode checkbox to. The setting is temporary and will not be considered for any later capture procedure. However, for the current session it overrides any definition from the overall settings profiles.

Click **NEXT** to proceed to the next wizard step.

## Options

**Note:**

The *Options* dialogue is restricted to the Capture Wizard's Expert Mode. The Wizard Mode is defined within the Settings section.

Within the **Options** step, there are two major groups of capture process settings that users are able to manipulate in this specific wizard cycle:

- **Exclusions**

The set of available exclusion definitions is changed by browsing for an alternative Exclusions configuration directory. Once the user has chosen the desired directory, the available exclusions are displayed at the bottom of the screen in the selection area on the left side. If the available exclusions do not match the current requirements, use the **ADD** or **EDIT** functionality to prepare decent exclusion definitions. Use the arrow controls to move exclusion definitions

---

from the available to the active list and vice-versa.

Excluded objects will be recorded from the capture machine during the snapshot generation, and later on marked as excluded within the capture project file. Target packages built from the project file will not contain excluded items.

- **Scan settings**

These options define which parts of the local capturing machine will be observed during the snapshot generation. If the application is to be installed on the C: drive, it is essential to activate scanning for this drive.

If the registry hives affected by the installation and configuration are well known, it is recommended to reduce the scanned set to the actual hives affected. Activate all options if the installation effects on the registry are not exactly known.

The definition of the scan settings is a tradeoff between scan speed and level of detail for the resulting **Delta** file. Therefore, the standard settings should cover a wider base, which can be narrowed down by experienced users in the expert mode during the individual wizard executions.

Click **NEXT** to proceed to the next wizard step.

## First Snapshot Configuration



**Note:**

The *First Snapshot Configuration* dialogue is restricted to the Capture Wizard's Expert Mode. The Wizard Mode is defined within the Settings section.

Within the **First Snapshot Configuration** step, there are two basic options packagers may decide among: They may either request the recording of a new snapshot, or re-use an already saved snapshot.

To actually reuse an existing snapshot, users have to activate the **Use existing snapshot** checkbox, and browse for the snapshot file that has to be used. Once selected, the snapshot is displayed with basic information, such as file size and creation time.

Click **NEXT** to proceed to the next wizard step.

## First Snapshot



**Be aware:**

The local machine has to be prepared with all required prerequisites before the next step is initiated since the initial snapshot capturing starts immediately.

**Note:**

The **First Snapshot** dialogue will not be displayed if a saved snapshot has been selected as capture data source.

---

During this wizard step, RayPack automatically scans the defined system areas and generates an *initial snapshot* file. The temporary snapshot files are saved into the directory that was defined within the settings area's snapshot tab of the currently active profile.

Depending on the system complexity and performance level, the capture process might take a while to finish.

Click **NEXT** to proceed to the next wizard step.

## Install Application

The Application to capture field is already filled in with the main installer that was defined within the Project Settings step. Before the **EXECUTE** button is used to start the actual setup routine, it is possible to add command-line arguments to manipulate the setup, e.g., when defining INI files to apply to a `.exe` based setup, or changing the `INSTALLDIR` property value for `.msi` related captures.

Hit the **EXECUTE** button to start the setup routine of the application to capture. Every effect the setup grabs on the capture machine is reflected in the later second snapshot, as long as it belongs to an area that is covered with the scan settings defined earlier.

Once the installation is run, the user has two basic proceeding options; either restart the capture machine if the installation instructions demand so, or go ahead to the next step. If a restart is required, RayPack is automatically launched again, with the Capture Wizard for the current project activated and navigated according to the relevant step.

**Note:**

If the Capture Wizard's Expert Mode is active, the newly created snapshot may be saved to disk as `.rcs` file.

---

Click **NEXT** to proceed to the next wizard step.

## Configure Application

**Note:**

The **Configure Application** dialogue is restricted to the Capture Wizard's Expert Mode. The Wizard Mode is defined within the Settings section.

---

Some applications require additional settings and configurations after the actual setup is completed. These should be defined within the installation instructions. The Experts Mode **Configure Application** wizard step is the appropriate time to execute these tasks.

Once the application is configured, the user has two basic proceeding options; either restart the capture machine if the installation instructions demand it, or go ahead to the next step. If a

restart is required, RayPack is automatically launched again, with the Capture Wizard for the current project activated and navigated according to the relevant step.

Click **NEXT** to proceed to the next wizard step.

## Second Snapshot Configuration



**Note:**

The **Second Snapshot Configuration** dialogue is restricted to the Capture Wizard's Expert Mode. The Wizard Mode is defined within the **Settings** section.

Within the **Second Snapshot Configuration** step, there are two basic options packagers may decide among: They may either request the recording of a new snapshot, or reuse an already saved snapshot.

To actually re-use an existing snapshot, users have to activate the **Use existing snapshot** checkbox, and browse for the snapshot file that has to be used. Once selected, the snapshot is displayed with basic information, such as file size and creation time.

Click **NEXT** to proceed to the next wizard step.

## Second Snapshot



**Be aware:**

All required installation and configuration tasks have to be fulfilled before this step is initiated since the second snapshot capturing starts immediately.

At this point, RayPack executes the same operations that were run for the capturing of the first snapshot. After its generation, the resulting temporary second snapshot is immediately compared to the initial one. RayPack transforms the Delta between the observed systems states into a project file in the `.rcp` format. The file is available in the project directory defined in the wizard step **Project Settings**.



**Note:**

If the second snapshot has been selected from an existing snapshot file resource, RayPack does not capture new information during this step, but simply compares the first and second snapshot to calculate and generate the delta file required for further packaging tasks.

The next wizard step is initiated automatically after the second snapshot generation has successfully completed.

## Finished

Click **FINISHED** to close the wizard dialogue.

To edit the newly created `.rcp` file, use the **OPEN** button from the action bar or click the open

---

capture project tile from the RayPack dashboard.

## Edit a Capture Project File

Opening a capture project file (.rcp file extension) launches the capture editor. Within this editor environment, some basic package properties can be edited to allow for adjustment of captured contents before the capture project is transformed into another package or project format.

The following information is about property manipulation and editing:

- **Package**

Set basic application properties such as Name, Publisher, Version and Language.

- **Files & folders**

Captured files and folders are displayable in a tabular or tree mode, allowing detailed inclusion and exclusion definitions.

Objects that are marked by red font color are currently excluded, and are not taken into account when a target file type is built from the project.

Excluding a folder has a downwards recursive action, meaning all sub-folders and files below will be excluded. Including a folder has an upwards recursive action, where including a leaf automatically includes all direct parent folders. However, including a folder does not automatically include the sub-folders and files within this folder.

List and tree views within RayPack usually support multi-selection functions. Use the control key on the keyboard to select several items at the same time. The info-column on the right hand side of the application window displays available group operations.

- **Registry**

The inclusion and exclusion system used for files and folders is also applied for registry management. Objects that are marked by red font color are currently excluded, and are not taken into account when a target file type is built from the project. The same rules for automatic downwards and upwards status effectiveness are applied.

- **Shortcuts**

Use the **IMPORT** button above the shortcut list to add a shortcut from the local capture machine to the capture project, such as Icon, title, target, etc. Pre-existing shortcuts displayed here can be excluded. Right-click the shortcut and select exclude from the context menu or use the **EXCLUDE** button, available from the info column at the right hand side. As previously explained, excluded items are marked by their red font color.

- **Permissions**

Permission changes for the registry and on files and folders that were committed during the setup or configuration phase of the capture wizard are listed within this section. Select one or more items from each list to exclude the entry from further consideration.

Permissions can only be changed from the captured status. To manipulate additional permission settings, build an MSI or RayPack project file from the capture project file and use the advanced options of the PackDesigner.

Registry permissions, as well as files and folders, can be exported to SecEdit inf files, which makes the security settings easily transferable.

Exclude the following items to keep them from taking effect in continuative package formats.

- **System Resources**

- **Services**

- If the setup or configuration activities caused changes to the set of services ran on the capture machine, they would be displayed on this list. Once a service is selected from the list, its properties can be edited within the info column on the right hand side. Options such as the service start and event type are changeable.

- **System variables**

- User and machine environment variables can undergo changes during a setup, and both types are considered by the capture wizard. If any system variable modification has been captured, its properties may be edited directly from within the list view of system variables. Select a variable from the list, click the column value that needs an update, and enter the new value.

## Build Packages from a Capture Project File

After the `.rcp` file has been edited to meet the required specifications, it is time to build the target package. To do so, open the Build view in the capture editor, and pick the desired format from the list of available targets. The active target format is marked by a background color change and a tick symbol in the upper right corner of the option area.



**Note:**

Once a target format is selected, other formats will become unavailable. Even though it is possible to generate several target formats during one generator run, not all make sense. If the target format selection needs to be changed, de-select the current target with a mouse-click so that it will enable the eligibility of other target options.

On the right hand side of the selected format, there is a configure button. If the build process needs to be adjusted compared to the standard format options (defined within RayPack's settings area), click this *configure* button to set the changes.

As always, the configuration changes are strictly local and apply exclusively on the target specific build process for the currently opened capture project. The standard profile settings are not affected by any changes made in the local project context. Please browse the **Knowledge Base**, or refer to the RayPack **Application Help** for details regarding available configuration options.

When the desired target format has been selected, and all configuration tasks are done, scroll down to the GENERATE button, click it to start the build process. RayPack displays a progress

indicator during the build process. Once the target packages are created, they are available from the project directory of the capture project.

## Troubleshooting

The various screws users can adjust for the customization of their PackRecorder results demand a multi-leveled troubleshooting strategy. Depending on the type of issue, different standard elimination procedures are at hand. Looking at the situations that may arise and answering the following questions should narrow the possible reasons for erroneous results to a trivial amount:

- **RayPack does not launch**

- Does the machine meet the hard- and software requirements to run RayPack?  
Please refer to the Release Notes for the current RayPack release to compare the actual system state with the requirements.
- Was RayPack licensed and activated correctly, and has the license period expired?

- **The capture wizard does not terminate successfully**

- Does the machine meet the hard- and software requirements to run RayPack?  
Please refer to the Release Notes for the current RayPack version to compare the actual system state with the requirements.
- Are all directory paths defined for the project and is the active RayPack profile correct and available from the capture machine?  
Please check the active profile settings, and try to access used directories and files.
- Are there processes or services running in the background that lockout required system areas?  
Make sure to terminate dispensable processes, services, and applications during the wizard execution.
- Are there multiple instances of RayPack running on the same capture machine?  
It is recommended to avoid instance issues by not trying to run several capture processes parallel. Make sure to save settings before the Capture Wizard is initiated.
- Does the current usage data log file contain messages regarding unavailable components or resources?  
Check the usage logs; they are usually generated within the `\RayPack\Logs\` directory of the user's profiles application data repository. (e. g., `C:\Users\<User>\AppData\Roaming\` on a Windows 7 machine)

- **The capture project file contains too much or not enough data**

- Do the standard profile settings fit the required scan areas? Were the adjustments made in the Expert's Mode wizard step "Options" correct with regards to the setup requirements?  
Please check the profile settings and re-run the Capture Wizard. Make sure to apply decent

---

exclusions and scan all required system areas.

- Are all directory paths defined for the project and is the active RayPack profile correct and available from the capture machine?  
Please check the active profile settings, and try to access used directories and files.
- Does the current usage data log file contain messages regarding unavailable components or resources?  
Check the usage logs; they are usually generated within the `\RayPack\Logs\` directory of the user's profiles application data repository. (e. g., `C:\Users\<User>\AppData\Roaming\` on a Windows 7 machine)

- **The file(s) generated from the capture project does not contain the expected data**

- Is the target format generation template available and accurate with the expected results?  
Open the required resource and check its values for correctness. Make sure to check the template that was actually used during the build process.
- Are all directory paths defined for the project and is the active RayPack profile correct and available from the capture machine?  
Please check the active profile settings, and try to access used directories and files.
- Does the current usage data log file contain messages regarding unavailable components or resources?  
Check the usage logs; they are usually generated within the `\RayPack\Logs\` directory of the user's profiles application data repository. (e. g., `C:\Users\<User>\AppData\Roaming\` on a Windows 7 machine)

In the case that the aforementioned dialogue and tips are unable to deliver a satisfying solution, refer to the **Application Help**, the **Knowledge Base** article library and community, or contact the RayPack support team via our [Support Panel](#). The vast experience and know-how of Raynet's packaging professionals is always available to handle complex situations.

## More Information

Please see the RayPack **Application Help** for further details or browse the **Knowledge Base** for additional articles and discussions.

# PackTailor Tutorial

## What is the PackTailor?

Packagers often have to go for the full distance, starting their job with repackaging, continuing with package design tasks and finalizing with validation and quality control. Nonetheless, there

are times when a little tweak of the given resources would pretty much do the trick. RayPack would not be recommended by professional packagers if it did not contain a perfect solution for this as well – the PackTailor.

Customizing an MSI may be time consuming and error-prone if there are many options in the installation database, including but not limited to serial numbers, paths, radio selections, feature states, license agreements etc.. With PackTailor, users are able to simply select changes they want to perform for the package.

By running a simulation of an MSI's UI sequence, packagers create a transform based on the set of available user input and selection options provided with the original MSI. Therefore, it is for example not necessary to know which property is required for serial number injection, PackTailor finds the right one, and records any changes made to it.

## What are the prerequisites for tailoring transforms?

Before any actual packaging activity, no matter if it is package design, transform tailoring, or repackaging, it is highly recommended taking a look at the settings section first. For details regarding settings, please refer to the **Settings Tutorial** within the Knowledge Base, or dive into RayPack's **Application Help** contents.

In order to create a decent MST from the tailoring process, the following preconditions have to be ensured:

- The resource file has to be a valid MSI package
- The base application should not be installed on the packaging machine
- Tailoring is not recommended for packages with complex launch conditions, system state analysis custom actions, or highly target machine related dependencies that cannot be recorded on the packaging machine.

## Tailor a transform

To initiate a new transform tailoring procedure, launch RayPack and click on the **Create a new project** tile on the dashboard. Creating a new transform is available by clicking on the last project type option.

Please refer to the section [Tailoring an MST for an MSI](#) to get a detailed description of the PackTailor wizard steps.

## Edit a tailored transform

Once the PackTailor wizard has been finished successfully, an MST file has been created. To manually open that transform with RayPack, apply one of the following options:

- Use the open tile on the RayPack dashboard and browse to the location where the MST is

saved. Click open.

- Use the FILE menu from the blue menu ribbon at the upper RayPack application window area. Select Open from the file menu, and browse to the location where the MST is saved. Click open.
- From a windows system browser, right-click the MST that should be opened with RayPack. Pick Edit with RayPack from the context menu. Please note that a new instance of RayPack will be initiated, additional to any already running ones.

RayPack will automatically ask to browse for the base MSI, and open it as well. Open the base MSI, and wait for the PackDesigner to load and analyze the given package resource information. As soon as the Your project overview page is displayed, your MST is ready for manipulation, validation and re-generation if required.

## Troubleshooting

The various screws users can adjust for the customization of their PackTailor results demand a multi-leveled troubleshooting strategy. Depending on the type of issue, different standard elimination procedures are at hand. Looking at the situations that may arise and answering the following questions should narrow the possible reasons for erroneous results to a trivial amount:

- **RayPack does not launch**

- Does the machine meet the hard- and software requirements to run RayPack?  
Please refer to the Release Notes for the current RayPack release to compare the actual system state with the requirements.
- Was RayPack licensed and activated correctly, and has the license period expired?

- **The PackTailor wizard does not terminate successfully**

- Does the machine meet the hard- and software requirements to run RayPack?  
Please refer to the Release Notes for the current RayPack version to compare the actual system state with the requirements.
- Are all directory paths defined for the project and is the active RayPack profile correct and available from the packaging machine?  
Please check the active profile settings, and try to access used directories and files.
- Is the base MSI itself valid towards the standard ICE rule set?
- If transforms were added to the tailoring process: Were the given in the right order and do they really belong to the active base MSI?
- Are there processes or services running in the background that lockout required system areas?  
Make sure to terminate dispensable processes, services, and applications during the wizard execution.

- Are there multiple instances of RayPack running on the same packaging machine?  
It is recommended to avoid instance issues by not trying to run several tailoring and / or capturing processes parallel. Make sure to save settings before the PackTailor Wizard is initiated.
- Does the current usage data log file contain messages regarding unavailable components or resources?  
Check the usage logs; they are usually generated within the `\RayPack\Logs\` directory of the user's profiles application data repository. (e. g., `C:\Users\<User>\AppData\Roaming\` on a Windows 7 machine)

- **The tailored transform does not contain the expected data**

- Did you deactivate some items from the UI sequence recording results? If so, depending objects may get lost as well. Please repeat the tailoring process and keep all recorded changes active for the MST creation if a clean result cannot be achieved with reduced result sets.
- If transforms were manually added to the tailoring process in the first place, make sure those are valid and contain the settings you expected to find in the resulting MST file. If the option to bundle all transform instructions within one combined MST was deactivated, the tailored transform will only include the changes made beyond the MSI-MST combination given as tailoring base.
- Are all directory paths defined for the project and is the active RayPack profile correct and available from the packaging machine?  
Please check the active profile settings, and try to access used directories and files.
- Does the current usage data log file contain messages regarding unavailable components or resources?  
Check the usage logs; they are usually generated within the `\RayPack\Logs\` directory of the user's profiles application data repository. (e. g., `C:\Users\<User>\AppData\Roaming\` on a Windows 7 machine)

- **RayPack does not launch PackDesigner after the PackTailor wizard has finished**

- Are several instances of RayPack running simultaneously? If so, make sure your packaging machine has enough capacity regarding processor performance and RAM to initiate another instance with the newly created file.
- Try to manually open the transform with RayPack. If this fails as well, the required resource may be out of reach due to broken network connections, changed user rights, and the like.
- Are all directory paths defined for the project and is the active RayPack profile correct and available from the packaging machine?  
Please check the active profile settings, and try to access used directories and files.
- Does the current usage data log file contain messages regarding unavailable components or

resources?

Check the usage logs; they are usually generated within the `\RayPack\Logs\` directory of the user's profiles application data repository. (e. g., `C:\Users\<User>\AppData\Roaming\` on a Windows 7 machine)

In the case that the aforementioned dialogue and tips are unable to deliver a satisfying solution, refer to the **Application Help**, the **Knowledge Base** article library and community, or contact the RayPack support team via our [Support Panel](#). The vast experience and know-how of Raynet's packaging professionals is always available to handle complex situations.

## More information

Please see the RayPack **Application Help** for further details or browse the **Knowledge Base** for additional articles and discussions.

## Need help?

### Request RayPack Support

Our Raynet support team gladly assists you on any question or issue you encounter regarding RayPack. Feel free to sign in and open incidents via our [Raynet Support Panel](#).

### Join the RayPack Community

The RayPack community resides within our Knowledge Base: <http://knowledgebase.raypack.net>. Once you signed up for access to the Raynet support panel, you automatically have access to the Knowledge Base, too.

You will surely come to a point where you would love to suggest a new feature for the future development of RayPack. Maybe you need to find some tips & tricks to hit your target right. The community is your place for discussing such topics, for sharing and expanding your own experience.

Step in contact with Raynet's professional packaging teams and learn how to polish your packaging activities to a level of highest quality standards. Since Raynet has years and years of experience in the packaging business, we know what to do, and how to do it. Don't row your boat alone when you have the chance to join our community for free.

### Contact your Raynet Sales Representative

Our sales team is the right contact for any license or edition question you might encounter. You would like to benefit from a professional RayPack training? Ask for dates and locations to find the fitting training occasion. You are highly welcome to step in contact via [sales@raynet.de](mailto:sales@raynet.de)



Raynet GmbH

Technologiepark 20  
33100 Paderborn, Germany

T +49 5251 54009-0

F +49 5251 54009-29

[info@raynet.de](mailto:info@raynet.de)

[support@raynet.de](mailto:support@raynet.de)

[www.raynet.de](http://www.raynet.de)